INTEGRATING EVENT DETECTION SYSTEM OPERATING CHARACTERISTICS INTO SENSOR PLACEMENT OPTIMIZATION

David B. Hart William E. Hart Sean McKenna
Sandia National Laboratories
Albuquerque New Mexico, USA
{dbhart, wehart, samcken}@sandia.gov

Regan Murray
US Environmental Protection Agency
Cincinnati, OH USA
Murray.Regan@epa.gov

Cynthia A. Phillips
Sandia National Laboratories
Albuquerque, NM USA
caphill@sandia.gov

Abstract

We consider the problem of placing sensors in a municipal water network when we can choose both the location of sensors and the sensitivity and specificity of the contamination warning system. Sensor stations in a municipal water distribution network continuously send sensor output information to a centralized computing facility, and event detection systems at the control center determine when to signal an anomaly worthy of response. Although most sensor placement research has assumed perfect anomaly detection, signal analysis software has parameters that control the tradeoff between false alarms and false negatives. We describe a nonlinear sensor placement formulation, which we heuristically optimize with a linear approximation that can be solved as a mixed-integer linear program. We report the results of initial experiments on a real network and discuss tradeoffs between early detection of contamination incidents, and control of false alarms.

Key words: sensor placement; sensor tuning, optimization; integer programming; water security

1 Introduction

The optimization of sensor placements is a key aspect of the design of contamination warning systems (CWSs). Online sensor stations collect data continuously and transmit it to a central database. Currently, water utilities use sensors that indirectly detect one or more contaminants. For example, typical water quality parameters like pH, chlorine, electrical

conductivity, oxygen-reduction potential, and/or total organic carbon can act as surrogate indicators for some contaminants.

At the central database/processing system, an event detection system (EDS) distinguishes periods of normal water quality from periods of anomalous water quality. The EDS tracks the levels of various sensor output over time, and determines when a pattern of activity might indicate a contamination. The EDS must have information about the normal behavior at each sensor location. Water quality parameters tend to vary significantly in water distribution systems due to normal changes in the operations (e.g. tanks, pumps, and valves), daily and seasonal changes in the source and finished water quality, and fluctuations in demands. For example, water quality variability may be higher near water sources and lower in residential neighborhoods.

We can tune the operating characteristics of the event detection software to account for water quality variability, and we generically view this tuning as setting a threshold parameter that determines whether water quality data is categorized as normal or anomalous. Selecting a tuning option determines a trade-off between false alarms and quality of detection. If a CWS operator would like highly sensitive sensors that will rapidly signal an alarm when a contaminant passes the sensor location, then that sensor location will likely have a higher false alarm rate. Similarly, if the CWS operator would like highly specific sensors that rarely signal false alarms, then these sensors will signal actual contamination incidents much later, or perhaps not signal them at all.

We describe an optimization formulation for sensor placement where we select locations accounting for their local water quality variability. Specifically, we select sensor locations based on their tuning parameters to minimize the expected impact over a set of potential contamination scenarios. We enforce an upper bound on the expected number of false alarms in a given unit of time (say a week, or a month), based upon a utility's tolerance for false alarms. This reflects a practical reality that false alarms generate work for the utility, and utility workers will not trust a sensor network that produces too many false alarms.

We solve this sensor-placement-and-tuning problem on an example real network with a heuristic optimization method. This experiment shows that having a low tolerance for false alarms can significantly constrain the options for placing sensors. In some cases where the tolerance is particularly low relative to the local false alarm rate at a single location, it may not even to possible to deploy all the sensor stations. The reduction in sensors counts and increase in detection failures caused by restrictive false alarm tolerances can significantly increase the expected impact from a suite of contamination events.

The remainder of the paper is organized as follows. In Section 2, we describe models for contamination movement, contamination detection, and sensor performance. We describe how sensor station sensitivity and specificity data can be represented as a set of receiver operator characteristic (ROC) curves. We describe how to use data from the Canary EDS (Hart and McKenna, 2009) to guide the selection of sensors in real-world distribution systems. This section also describes how we calculate the input to the specific mathematical model described in Section 3. In Section 4, we describe the design of our experiment. We give the results of that experiment in Section 5, and we conclude with a discussion of related

and future work in Section 6.

2 Modeling

In this section we describe our modeling assumptions and methods for generating input for our optimization formulations.

2.1 Contamination Transport and Impact Calculations

As in a series of papers beginning with (Berry et. al., 2004), we wish to place p sensors at junctions of a water distribution network to minimize the average impact of a suite of contamination incidents. Each contamination incident is associated with an injection point, time of day, length of injection, nature of the contaminant (type, concentration), and any other information pertinent to calculating its impact. For each incident in the contamination suite, we simulate the movement of the contamination through the network using EPANET (Rossman, 1999). EPANET's simulation algorithms assume a set of demand patterns that continuously cycle until the end of the simulation horizon. In typical simulations, demand patterns hold steady for an hour at a time and the pattern cycle repeats each day. In general, one can build incident suites that cover any foreseeable scenario based on, for example, day of the week, season, or special event. For each contamination event, EPANET reports concentration of contaminant at the network junctions over time and reports where and when water leaves the network at demand junctions. Thus we can determine where and when a sensor placed at a particular junction can observe the given event.

There are a number of metrics suitable for measuring the impact of a contamination event on a network or population (Hart and Murray, 2010). The primary measure of network impact is the number of pipe-feet exposed to the contaminant. Given information about the population distribution and the health effects of the contaminant, we can calculate the number of people exposed to, sickened by, or killed by contamination. The volume or mass of contamination removed from the network at demand junctions can be a surrogate for population metrics when there is insufficient information to compute the more precise measures. Although some metrics, such as number of failed detections, do not depend upon the time between release and detection, we will consider impact metrics that do increase the longer a contamination incident proceeds undetected. Thus, given a time t since the beginning of an event, we can compute the total impact to the network and/or population up through time t.

When contamination passes a sensor s_i at location i, the information it sends to the database may trigger an alarm. Suppose sensor s_i first detects significant contamination t_i minutes after the start of the contamination injection. The event detection system (EDS) may introduce some delay δ_i between time t_i and the time it signals a controller; see Section 2.2 for more details. The utility may then introduce a response delay δ_R to verify the contamination, perhaps through manual sampling. Let t_A be the earliest time after an injection when the utility can be reasonably sure of the contamination incident. That is,

$t_A = \min_i(t_i + \delta_i) + \delta_R$. We assume at time t_A there is no further impact from the contamination incident, because, for example, the utility contains the contamination or issues an alarm to stop water consumption with instant public response. For any given deployment of sensors and any given contamination incident, the location i where an observation of the contamination leads to the earliest alarm is called the *witness* for the event.

In this paper, we will assume for simplicity that the response delay $\delta_R = 0$ for all incidents and witnesses. Thus whenever we refer to delay, we mean the delay between the time the sensor detects an incident and the time the EDS signals the controller.

2.2 Detecting with Imperfect Sensor Systems

Current event detection systems (EDS) analyze standard water quality parameters for periods of anomalous behavior. These water quality signals (e.g., residual chlorine, pH, total organic carbon, etc.) serve as surrogates for a contaminant specific signal. CANARY (Hart and McKenna, 2009) is an EDS that does multivariate signal analysis in real time and provides a continuously updated probability of a contamination event to the water quality analyst.

CANARY employs a number of filtering algorithms to estimate the expected water quality value at any given time. CANARY considers the water quality values from a given sensor from time period to time period, where the length of the time period is generally about five minutes. CANARY learns a pattern of normal behavior for water quality at a given location by considering historical correlations. So, for example, if a particular water quality parameter tends to be higher in the morning than in the afternoon, CANARY will consider that when predicting normal ranges for morning vs afternoon measurements. If the value tends to be reasonably steady, then there may be a weighting to consider recent values more heavily than those in the more distant past.

To compute the expected value of a particular water quality measurement for the next time period, CANARY considers the recent behavior of the system. In particular, it considers the values of a water quality measure for each time period in a time window immediately preceeding the current time, usually the last day or two. CANARY has previously classified the measurement in each of these time periods as an *outlier* (abnormal), or consistent with normal values. CANARY computes a weighted average of all the normal values. That is, it drops all the outlier (abnormal) values when computing an expected value. The weightings in this average come from the known or observed correlations mentioned above. This gives an expected value for the measurement m and a standard deviation σ. CANARY updates the weighting it applies to each previous value when the time window shifts forward to compute the next expected value.

In each time period, CANARY computes the difference $(m - m')$ between this estimated (expected) water quality m and the measured water quality m'. The user specifies a threshold γ on this difference, given as a multiple of the standard deviation. Time steps at which $|m - m'| > \gamma\sigma$ are classified as outliers. Otherwise, the value m' is considered sufficiently consistent with background water quality. Note that an outlier can be in either direction, high or low.

Because a water network has some fluctuations in water quality measurements even when there is no contamination event, CANARY does not signal an alarm when it first observes an outlier. Instead, CANARY uses a binomial event discriminator (BED) to determine the probability that the pattern of outliers is signalling a contamination event (McKenna et al., 2007). The BED considers classification of the most recent time steps, typically the previous one to two hours, as a binomial process (like flips of a coin) where outliers are considered failures. This testing time window is much smaller than the time window used for computing expected water quality values. The probability of the water quality representing background conditions decreases according to the binomial model as the number of outliers within the moving window of recent time steps increases. More specifically, the probability p_f of having an outlier (failure) during a time when there is no contamination event depends upon the threshold γ described above. Assuming the measurement at each of the n_b time periods in the testing window is an independent trial, one can compute the probability of observing n_O outliers during n_b normal time periods (trials) using the standard binomial formula. This gives a probability P_b that the testing time window is observing background conditions. The probability of an event is $P_e = 1 - P_b$. When P_e is larger than an operator-defined threshold, CANARY signals an alarm.

Event detection systems like CANARY can make mistakes in two possible ways. They can signal an alarm when there is no real contamination (a false alarm), or they can fail to signal a real contamination event (a false negative). McKenna et. al. (McKenna et al., 2008) quantify EDS performance using receiver operating characteristics (ROC) curves (Collinson, 1998; Metz, 1978). The ROC curve estimates the tradeoff between the false alarm rate and the probability of detecting a true event. That is, it is a plot of false alarm rate (given as an expected number of false alarms per unit time) vs the probability of a detecting a true event; for example, see Figure 1. The ROC curve depends upon both the sensor location and the length of the testing time window n_b. Higher sensitivity within a detection system typically leads to increases in both the probability of detection and the number of false alarms. In CANARY, the choice of the thresholds described above determines a point on the ROC curve. That is, it selects a pairing of false alarm rate and false negative probability (which is 1 minus the probability of detection).

Although characteristics of ROC curves for sensors vary throughout a water distribution system, properties such as water quality variability appear to have a significant impact on these characteristics. In the rest of this paper, we assume that properties like water quality variability can be used to categorize network junctions into a small number of classes, each of which shares a ROC curve that represents sensor behavior within that class. Thus the EDS time delay δ_i described in section 2.1 is a delay δ_c associated with node class c, and the node class completely determines the appropriate ROC curve for any given location. However, the utility can use this ROC curve data to independently tune sensors at different locations, which is the focus of the next section.

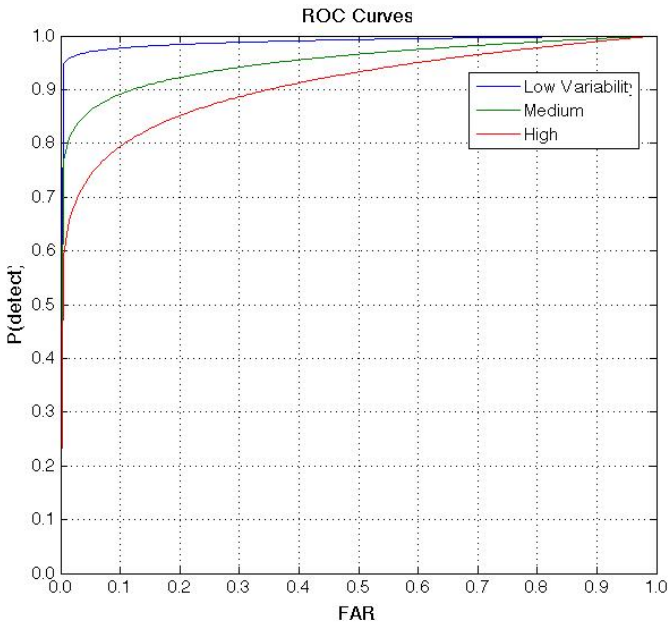


Figure 1: Illustration of a ROC curve, with three different curves that illustrate how water quality variability impacts sensor performance.

3 Problem Formulation

We now describe a mixed-integer programming (MIP) formulation for the sensor placement problem with sensor tuning. We are given a set L of locations where we can place sensors in a network. We can choose p locations to receive a sensor. Each location $i \in L$ has K_i tuning levels, based upon its junction class. Each tuning level corresponds to a point on the appropriate ROC curve, as described in Section 2.2. The junction class for location i also determines how long we expect to wait for the EDS to signal an alarm after the contaminant first touches location i, provided the EDS sucessfully recognizes the contamination incident.

We are also given a set $\mathcal{A}$ of contamination incidents. Each incident $a \in \mathcal{A}$ has a weight α_a, which could be the probability the incident occurs conditioned on some incident occuring. For our experiments, we assume all incidents are equally likely, so $\alpha_a = 1/|\mathcal{A}|$ for all incidents a. Let $\mathcal{L}_a$ be the set of locations touched by contamination incident a. A sensor placed at any location $i \in \mathcal{L}_a$ might witness incident a.

If a sensor at location i witnesses incident a, then incident impact is the total impact at the time the EDS signals an alarm, which includes the detection and response delays described above. Let d_{aik} be the impact if incident a is witnessed by a sensor at location i set to tuning level k. Set $\mathcal{L}_a$ contains a special dummy junction q for all incidents a. The dummy represents a failure to detect the incident. Thus the dummy serves as a witness for any incident that is not witnessed by a real sensor. The dummy has an impact equal to the

total impact if the incident is not detected until there are other indications from something outside the sensor network, such as people becoming ill and seeking medical treatment. We set the time horizon on the EPANET simulations based on our estimate of when such external indicators will be recognized. Thus the impact for the dummy is the total impact at the end of the simuation.

Our modeling of sensor failures is based upon that in (Berry et. al., 2009), which considered failures of untunable sensors. The choice of tuning level k determines a false alarm rate f_{ik}. This is the expected number of false alarms per unit time generated by a sensor placed at location i tuned to level k. Due to linearity of expectation, the total number of false alarms per unit time in the whole sensor network is the sum of these rates for all locations receiving sensors. We require this total rate to be below a utility-defined tolerance F.

The choice of tuning level k also determines a false negative probability r_{ik}. As described in (Berry et. al., 2009), the probability that a sensor witnesses an incident when sensors can fail to detect an incident is a naturally nonlinear problem. For incident a, consider the locations in $\mathcal{L}_a$ sorted by increasing alarm time if the sensor correctly detects the incident. A sensor at location i is a witness only if all the sensors that could have suceeded earlier (its predecessors) fail, and it succeeds. If all detections are independent events, the witness probability for location i is a product of the failure probabilities for all predecessor sensors and the success probability for location i. The natural mathematical program given in (Berry et. al., 2009) involves the product of sensor-placement decision variables, and is therefore nonlinear.

However, (Berry et. al., 2009) give a linear approximation for this witness probability, called *one imperfect witness*, which for their experiments gave a reasonable approximation to the best solutions using the more complex formulation, but required far less computation. For a given incident a, the candidate witness locations are those in $\mathcal{L}_a$ (i.e. those touched by contamination in incident a) that receive a sensor. In the one-imperfect-witness formulation, the probability of witnessing an event is not partitioned among multiple candidates. Instead the witness probability is divided between a single candidate and the dummy. This corresponds to picking a single sensor to attempt to witness the event. If it fails, then we assume no sensor succeeds. In practice, the other sensors have a chance to witness the event, so this approximation pessimistically calculates expected impact. However, we can calculate the actual impact based on the more complex impact calculations sketched above.

The one-imperfect-witness model is equivalent to the model where sensors are perfect, with adjusted impact values. When all candidates detect an incident with probability 1, the sensor that sees an incident first will have lowest impact and will be the witness. When impacts reflect failure probabilities, then the first potential witness may not be the best single choice if it has a high failure probability.

Let d_{aq} be the impact for the dummy location, representing no detection, for incident a. Then the adjusted impact value d'_{aik} for a sensor at location i with tuning level k is $d'_{aik} = (1 - r_{ik})d_{aik} + r_{ik}d_{aq}$.

The following MIP model provides a one-imperfect-witness formulation that allows for choice of sensor tuning. This MIP is similar to the perfect-sensors formulation from (Berry

et. al., 2004), and it extends the one-imperfect-witness formulation from (Berry et. al., 2009) to include choices on tuning parameters as well as a bound on the false alarm rate.

$$\text{(SPT)} \quad \text{minimize} \quad \sum_{a \in \mathcal{A}} \alpha_a \sum_{i \in \mathcal{L}_a} \sum_{k \in K_i} d'_{aik} x_{aik}$$

$$\text{where} \quad \begin{cases} \sum_{i \in \mathcal{L}_a} \sum_{k \in K_i} x_{aik} = 1 & \forall a \in \mathcal{A} \\ x_{aik} \leq s_{ik} & \forall a \in \mathcal{A}, i \in \mathcal{L}_a, k \in K_i \\ \sum_{i \in L} \sum_{k \in K_i} s_{ik} \leq p & \\ \sum_{k \in K_i} s_{ik} \leq 1 & \forall i \in L \\ \sum_{i \in L} \sum_{i \in K_i} f_{ik} s_{ik} \leq F & \\ s_{ik} \in \{0, 1\} & \forall i \in L, k \in K_i \\ 0 \leq x_{aik} \leq 1 & \forall a \in \mathcal{A}, i \in \mathcal{L}_a, k \in K_i \end{cases}$$

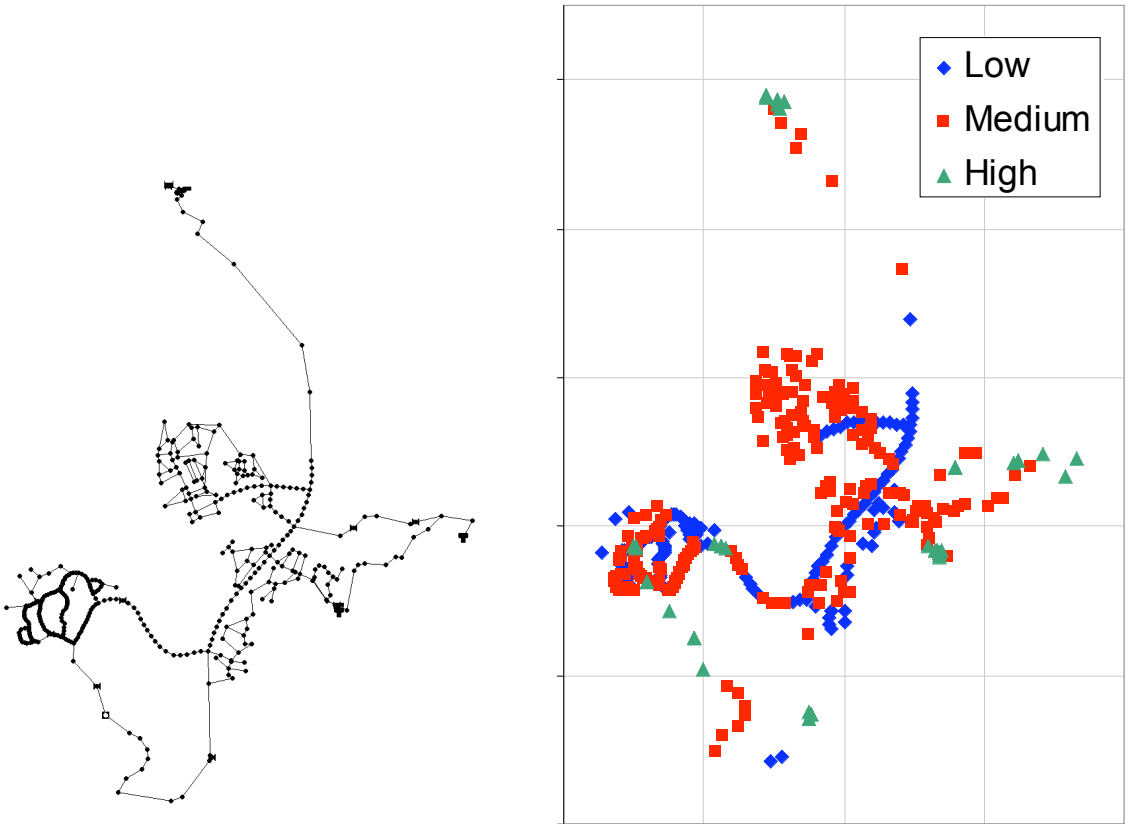
In this model, variable s_{ik} is 1 if we place a sensor on location i tuned to level k. Variable x_{aik} is the probability that a sensor a location i tuned to level k witnesses event a. The objective function minimizes the expected impact using the one-imperfect-witness approximation. The first set of constraints ensures that each event is witnessed by either a sensor or the dummy. The second set of constraints enforces that only real sensors can serve as witnesses for any event. The third constraint enforces the sensor budget: we can place no more than p sensors in the network. The fourth set of constraints picks a tuning level for each real (placed) sensor. Each location can host at most one sensor. The fifth constraint limits the total false alarm rate for sensors that are placed in the network. The remaining constraints require the sensor-placement variables to be binary and require the witness variables to be probabilities (i.e. to have values between 0 and 1).

4 Experimental Design

In our experiments we performed sensor placement on network 1 from the Battle of the Water Sensor Networks (Ostfeld et. al., 2008), augmented with extra junctions on some pipes. This network includes junctions that allow placement of sensors in the middle of some pipes. This network has 407 junctions, 448 pipes, 106 non-zero-demand junctions, 3 pumps, and 8 valves. Figure 4a shows this network.

Because we expect junction-class information to be fairly coarse, we have partioned the junctions into three classes based on low, medium, and high flow variability. Figure 4b shows the distribution of the junction classes within the network.

It is still an open research question what properties of a junction most affect the ability of an EDS to perform correctly. We believe that "high variability" regions are at greater risk for failures, but it is not clear what is varying. In this experiment, we use the variability of flow rates through a given junction as a proxy for event detection system performance at that junction. Higher variability in flows is associated with decreased performance of the EDS. EPANET does not calculate flows through junctions, so we used the variability of flows in pipes connected to each junction in the calculation. Flow variability for the ith junction, V_i, is the weighted sum of the standard deviations of the flows within each of the η_i pipes



(a) (b)

Figure 2: The network used in our experiments (a) The Network toplogy (b) Color coding of the flow variability classes.

connected to that junction. The variabilities are calculated as a proportion of the total flow through the junction:

$$V_i = \frac{\sum_{j=1}^{\eta_i} \sigma_j Q_j}{\sum_{j=1}^{\eta_i} Q_j},$$

where Q_j and σ_j are the absolute value of the volumetric flow rate and the standard deviation in the absolute values of the flow rates, respectively, within the jth pipe adjacent to junction i. For each pipe, we calculated the values of σ_j across all time steps in the model. Figure 4 shows the number of junctions with flow variability in various ranges. We consider all junctions i with $\log_{10} V_i < 1.6$ gallons per minute (gpm) to be "low" variability. We consider all junctions i with $1.6 \leq \log_{10} V_i < 3.5$ gpm to have "medium" variability. Finally, we consider all junctions i with $\log_{10} V_i \geq 3.5$ gpm to be "high" variability.

We generated three ROC curves: one for each of the junction classes. We used hypothetical performance characteristics for event detection in our experiments, but they are based on the application of CANARY in several municipal systems within the US. Table 1 gives the expected EDS detection delay for each junction class and three combinations of false

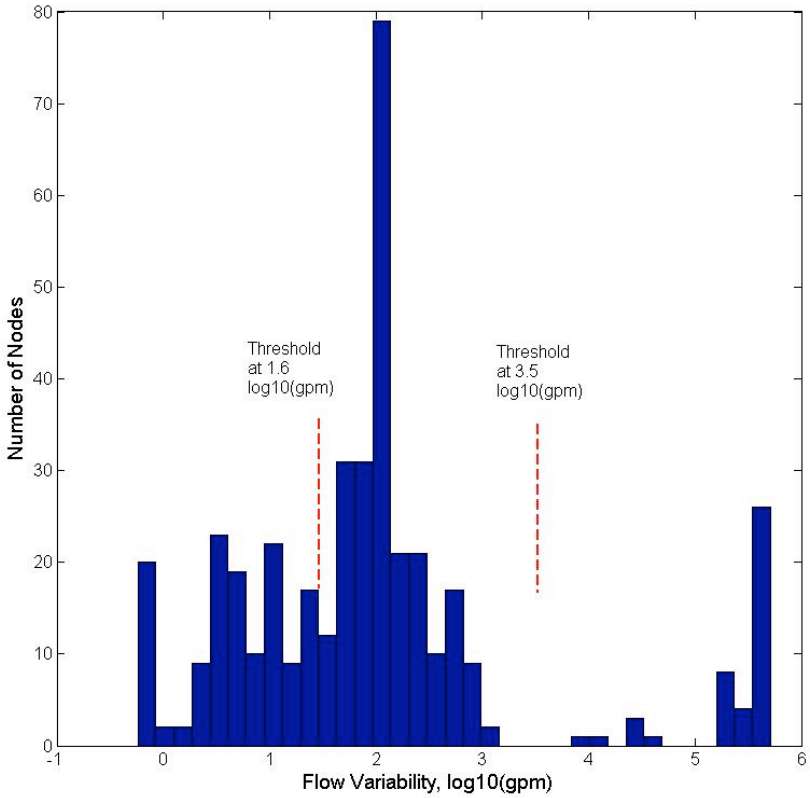


Figure 3: The flow variability in our example network. Each bar represents the number of junctions with log(flow variabilty) in a given range, measured in gallons per minute.

alarm rate (FAR) and false negative probability. We estimated the detection delay from the BED. For junctions with higher flow variability, the BED uses a larger test time window, so there are more time steps (the binomial number of trials) over which information on outliers is aggregated prior to sounding an alarm. These delays are the time it takes CANARY to sound an alarm, given that a true event is being detected.

Our incident suite involves one contamination incident per nonzero-demand junction. It is a 24-hour injection of a large number of biological cells. Our demand patterns repeat after 96 hours. We ran the EPANET simulations for 192 hours with a 5-minute time step size for both reporting and quality.

Our sensor placement experiments consider the placement of 5, 10, 20, or 40 sensors. We computed the extent of contamination (ec) impact measure, which computes the total pipe feet of the water distribution network exposed to contamination. We constrain the total false alarm rate to be either 1 per day, 1 per week, 1 per 2 weeks, or 1 per 4 weeks.

The SPT optimizer is configured to use a bound on the number of sensors. For low false

Table 1: Values extracted from Receiver Operating Characteristics (ROC) curves. FA is false alarm and FAR is false alarm rate, the expected number of false alarms per day. The next-to-last column gives the number of false alarms a utility can expect in a day for a sensor network containing 10 sensors, each with the same flow variability and tuning choice. The last column gives the expected time between false alarms for this same type of network. This is simply the inverse of the preceeding column.

Flow Variability at Junction	FAR in FA per day-sensor	Probability of Failure to Detect	Conditional Detection Delay in time steps (minutes)	Network-wide FAR per day assuming 10 sensors	Average Time between FAs in days
Low	0.010	0.045	5(25)	0.10	10
	0.025	0.0362	5(25)	0.25	4
	0.050	0.0295	5(25)	0.50	2
Medium	0.010	0.2057	7(35)	0.10	10
	0.025	0.1684	7(35)	0.25	4
	0.050	0.1391	7(35)	0.50	2
High	0.010	0.369	9(45)	0.10	10
	0.025	0.3085	9(45)	0.25	4
	0.050	0.2589	9(45)	0.50	2

alarm rate tolerances, the best solution may not use all the sensors the utility has budgeted, depending upon our choices of points on the ROC curve (our tuning options). In fact, there may be no feasible solutions that use all available sensors if the per-sensor false alarm rate is too high relative to the tolerance for the total false alarm rate. In that case, even when all sensors are deplayed to low-variability zones, there are too many expected false alarms.

5 Results

Table 2 shows our experimental results from optimizing SPT with different limits on the number of sensors (p) and the expected total false alarm rate (FAR). We denote this tolerance limit on the FAR by F in the SPT model. We call the user-specified maximum number of sensors or tolerance for number of false positives (upper) *bounds* because no feasible solution can exceed p sensors or F total false alarms per day.

There are several noteworthy trends in these results. First, as the tolerance for false alarms decreases, the number of sensors in an optimal configuration generally decreases; the only exception is in the first group of experiments, where the nonmonotonicity can be attributed to the discrete nature of the choices being made and the specific false alarm rates for the different tuning levels. This trend confirms that the bound on FAR is often constraining in these sensor placement examples.

Another trend is that the distribution of sensors tuned with high, medium and low FAR (sensitivity) shifts from high to low as the tolerance on FAR decreases. This follows as one would expect from the mathematical representation of the total FAR constraint. Tight bounds on total FAR require the use of sensors that with low sensitivity, and hence low false-alarm rates.

Similiarly, as the tolerance on FAR decreases, the expected contamination impact with the optimized sensor placement increases. This reflects the fact that the sensors being deployed are less sensitive and thus the expected utility for sensors is lower. The impact of the FAR tolerance on the expected contamination impact increases as the number of sensors allowed increases. This is because the highest FAR tolerance we consider, 1.0, does not tightly constrain the sensor designs for the smaller number of sensors (5 and 10). Thus, there is flexibility in these designs to leverage more sensors to reduce the expected contamination impact.

These results confirm that the bound on FAR is a dominant factor in the types of designs that are generated when optimizing SPT. For low FAR tolerances, the same solutions are generated regardless of the bound on the number of sensors. For high FAR tolerances, there is more flexibility in the design space, and thus the number of sensors allowed can have a big impact on the utility of the sensor network design.

6 Discussions

In this section, we discuss previous related work and some possible future directions.

Table 2: Experimental results from solving (SPT) with different numbers of sensors and limits on the total false alarm rate. FAPW is the number of false alarms tolerated per week. F is the tolerance (maximum) on the expected number of false alarms per day, simply 1/7 the preceeding column. The actual expected number of false alarms per day, calculated from the actual sensor placement, is given in expected false alarms per day. The impact is the one-imperfect-witness value, the SPT objective, not the actual expected impact for the sensor placement.

Num Sensors		Sensor Levels			False Alarm Rate			Mean Impact
Max	Used	high	med	low	FAPW	F	Actual	
5	5	5	0	0	7	1	0.25	6698.68
5	3	1	1	1	1	0.143	0.135	6846.55
5	5	0	1	4	0.5	0.0714	0.065	7161.67
5	3	0	0	3	0.25	0.0357	0.03	8838.34
10	10	10	0	0	7	1	0.5	4973.57
10	10	1	0	9	1	0.143	0.14	5510.71
10	7	0	0	7	0.5	0.0714	0.07	6559.56
10	3	0	0	3	0.25	0.0357	0.03	8838.34
20	20	20	0	0	7	1	1.00	3895.86
20	14	0	0	14	1	0.143	0.14	5127.66
20	7	0	0	7	0.5	0.0714	0.07	6559.56
20	3	0	0	3	0.25	0.0357	0.03	8838.34
40	39	12	7	20	7	1	1.00	3452.34
40	14	0	0	14	1	0.143	0.14	5127.66
40	7	0	0	7	0.5	0.0714	0.07	6559.56
40	3	0	0	3	0.25	0.0357	0.03	8838.34

This is the first work we are aware of that considers sensor tuning options as part of the sensor placement decisions. However, researchers are beginning to consider sensor failures in general. Our models for placement of imperfect sensors with tuning was based upon the work in (Berry et. al, 2009) for untunable imperfect sensors. The survey of sensor placement research in (Hart and Murray, 2010) also mentions this work: "(Preis and Ostfeld 2008) use a detection redundancy method to increase reliability. (Krause et al. 2008) allow for mechanical sensor failures."

There are many possible way to assign junctions to classes based upon the difficulty a EDS may have in classifying events at that location. The best way would be to perform a direct study of water quality variability under normal and abnormal circumstances using, for example, the capabilities in EPANET MSX (multi-species extension). Otherwise, one could consider using other measures such as the raw speed of water movement, the amount of mixing at a junction, and the number and nature of changes in water flow direction.

Varying the size of the testing time window in the EDS can provide additional tuning

flexibility. In general, any important parameter or algorithmic choice for the EDS, one that has a large effect on EDS behavior, might be exposed as a sensor network design choice.

For a fixed choice of testing time window size, we could better model the expected time of an alarm to the operator by considering the EDS decision at each time step. For a testing window of size n_b, there must be at least some minimum number of outliers $n_{\min}$ in that window for an EDS like CANARY to signal an alarm. The earliest this could happen is $n_{\min}$ steps after the contamination first arrives. Depending on the size of $n_{\min}$ relative to n_b, the EDS will be more likely to signal an alarm as the event proceeds. For example, the EDS might have a 50% chance of missing an event at the earliest possible time, but it might have a 90% chance of detecting it within the next n_b steps. We can use this better estimate to compute a better estimate of the expected impact for any given witness for any given incident.

It would be valuable to find a reliable method for computing good lower bounds for the expected impact of the sensor placement problem in its full nonlinear form. This will allow better evaluation of the quality of approximate solutions. As the technology for solving mixed-integer nonlinear programs improves, we may able to solve this full nonlinear problem directly.

Acknowledgements

Sandia is a multiprogram laboratory operated by Sandia Corporation, a wholly owned subsidiary of Lockheed Martin company, for the United States Department of Energy's National Nuclear Security Administration under Contract DE-AC04-94-AL85000. The U.S. Environmental Protection Agency through its Office of Research and Development partially funded and collaborated in the research described here under an Interagency Agreement with the Department of Energy. It has been subjected to the Agency's review and has been approved for publication as an EPA document.

References

[1] J. Berry, R. D. Carr, W. E. Hart, V. J. Leung, C. A. Phillips, and J.-P. Watson, *Designing contamination warning systems for municipal water networks using imperfect sensors*, Journal of Water Resources Planning and Management, 135 (2009), pp. 253–263.

[2] J. Berry, W. E. Hart, C. A. Phillips, and J. Uber, *A general integer-programming-based framework for sensor placement in municipal water networks*, in Proc. World Water and Environment Resources Conference, ASCE, Reston, VA, 2004.

[3] P. Collinson, *Of bombers, radiologists, and cardiologists: Time to ROC*, Heart, 80 (1998), pp. 215–217.

[4] D. B. Hart and S. A. McKenna, *Canary user's manual version 4.2*, Tech. Rep. 600/R-08/040A, United States Environmental Protection Agency, Nov 2009.

[5] W. E. Hart and R. Murray, *A review of sensor placement strategies for contamination warning systems in drinking water distribution systems*, J Water Resources Planning and Management, (2010). (to appear).

[6] A. Krause, J. Leskovec, C. Guestrin, J. VanBriesen, and C. Faloutsos, *Efficient sensor placement optimization for securing large water distribution networks*, Journal of Water Resources Planning and Management, 134 (2008), pp. 516–526.

[7] S. A. McKenna, D. B. Hart, K. A. Klise, V. A. Cruz, and M. P. Wilson, *Event detection from water quality time series*, in Proceedings of the World Water and Environmental Congress, 2007.

[8] S. A. McKenna, M. Wilson, and K. A. Klise, *Detecting changes in water quality data*, Journal of the American Water Works Association, 100 (2008), pp. 74–85.

[9] C. Metz, *Basic principles of ROC analysis*, Seminars in Nuclear Medicine, VII (1978), pp. 283–298.

[10] A. Ostfeld, J. G. Uber, E. Salomons, J. W. Berry, W. E. Hart, C. A. Phillips, J.-P. Watson, G. Dorini, P. Jonkergouw, Z. Kapelan, F. di Pierro, S.-T. Khu, D. Savic, D. Eliades, M. Polycarpou, S. R. Ghimire, B. D. Barkdoll, R. Gueli, J. J. Huang, E. A. McBean, W. James, A. Krause, J. Leskovec, S. Isovitsch, J. Xu, C. Guestrin, J. VanBriesen, M. Small, P. Fischbeck, A. Preis, M. Propato, O. Piller, G. B. Trachtman, Z. Y. Wu, and T. Walski, *The battle of the water sensor networks (BWSN): A design challenge for engineers and algorithms*, J Water Resource Planning and Management, 134 (2008), pp. 556–568.

[11] A. Preis and A. Ostfeld, *Genetic algorithm for contaminant source characterization using imperfect sensors*, Civil Engineering and Environmental Systems, 25 (2008), pp. 29–39.

[12] L. A. Rossman, *The EPANET programmer's toolkit for analysis of water distribution systems*, in Proceedings of the Annual Water Resources Planning and Management Conference, ASCE, Reston, VA, 1999. Available at http://www.epanet.gov/ORD/NRMRL/wswrd/epanet.html.